Topics in Deployable ML:

Min-Max Optimization I

(Lecture 8)

Costis Daskalakis

Motivation for this Lecture

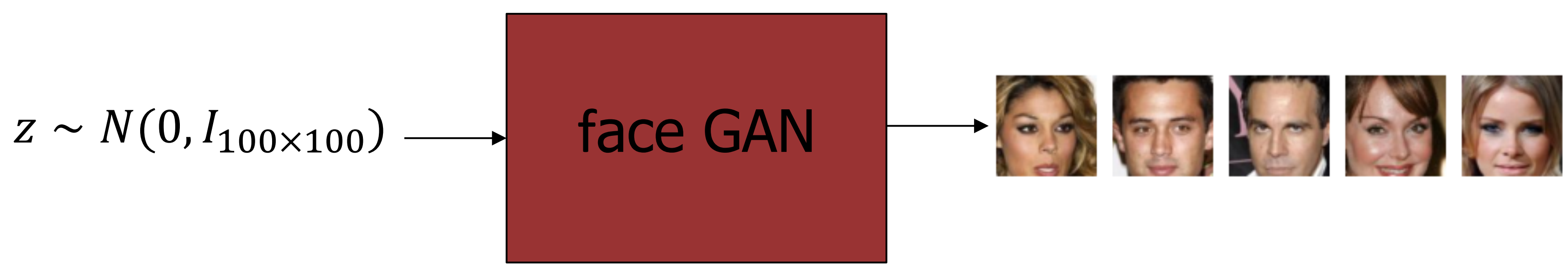
Minimization is to current AI

what min-max optimization is to future AI

- **Minimization:** AI agent is learning in a stationary environment
- **Min-max optimization:** AI agent is learning in a *changing* environment
- Why changing?
  - Because noise/adversaries poison or corrupt the data [*c.f. lecture 4*]
  - Because the agent is optimizing against another agent with conflicting interests
  - Because the agent wants to enforce constraints on the learning outcome, e.g. GANs, private release of data etc.

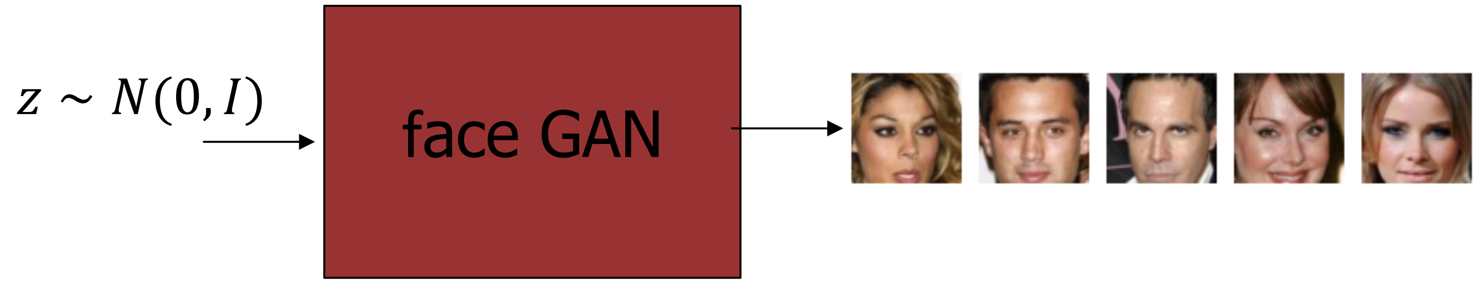
Generative Adversarial Networks

- Algorithms mapping white noise to random high-dimensional objects with structure:



- If you want, what human imagination does (presumably)
- Trained using samples (e.g. faces) from true high-dimensional distribution with structure (e.g. natural face images)

E.g. Wasserstein GAN
[Arjovsky-Chintala-Bottou'17]

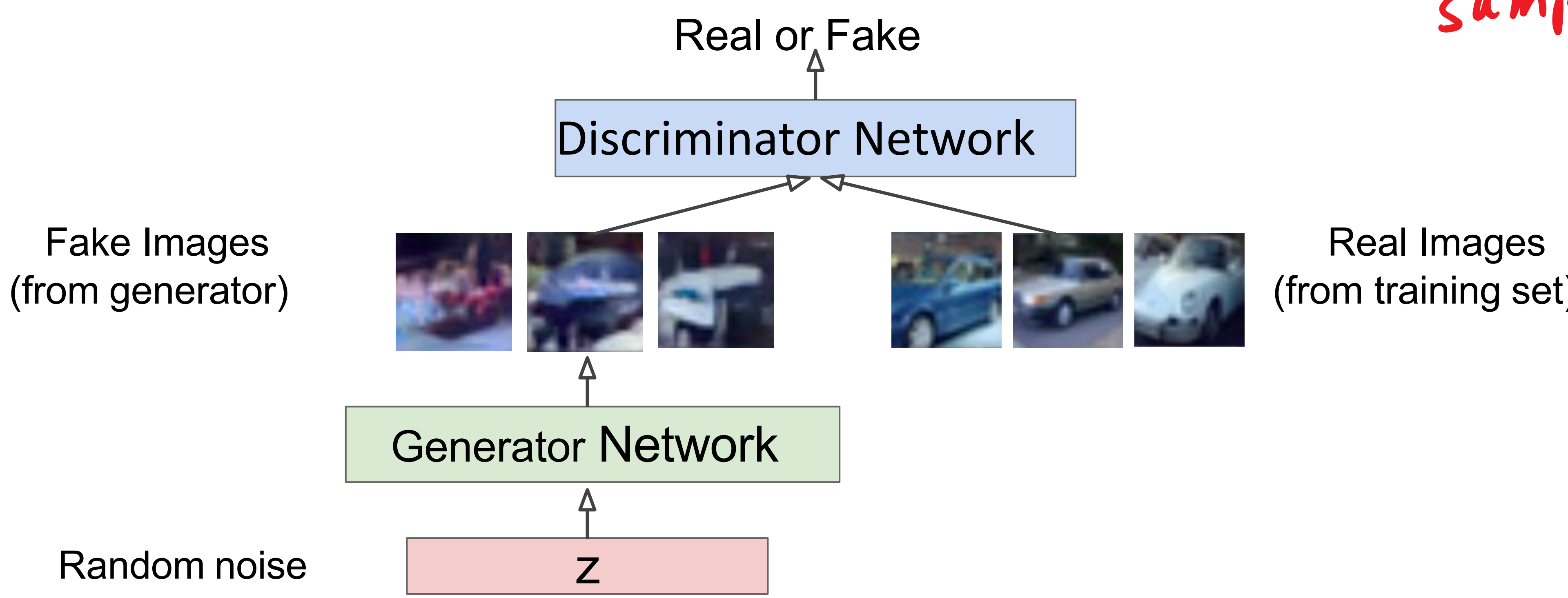


- A ***game*** between a *Generator* deep NN, w/ parameters θ_g, and a *Discriminator* deep NN, w/ parameters θ_d:

$$\inf_{\theta_g} \sup_{\theta_d} \left(\mathbb{E}_{X \sim F}[D_{\theta_d}(X)] - \mathbb{E}_{Z \sim N(0,I)} \left[D_{\theta_d} \left(G_{\theta_g}(z) \right) \right] \right)$$

sample from real F

halucinated sample



- **Training**: generator and discriminator run gradient descent and ascent respectively to update their parameters θ_g, θ_d; expectations are approximated by finite sample averages
- even ignoring expectation approximation errors, will paired gradient descent/ascent dynamics converge? to what?

Menu

- **Motivation**
- Min-Max Theorem
- No-Regret Learning and Online Convex Optimization
- Back to Min-Max Optimization

Menu

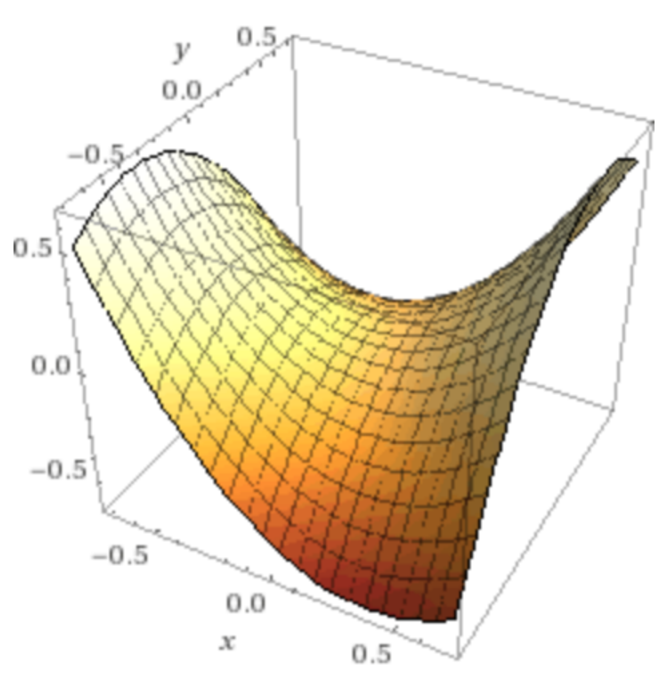
- **Motivation**
- **Min-Max Theorem**
- No-Regret Learning and Online Convex Optimization
- Back to Min-Max Optimization

The Min-Max Theorem

- **[von Neumann 1928]:** If $X \subset \mathbb{R}^n, Y \subset \mathbb{R}^m$ are compact and convex, and $f: X \times Y \to \mathbb{R}$ is convex-concave (i.e. $f(x, y)$ is convex in x for all y and is concave in y for all x), then

$$\min_{x \in X} \max_{y \in Y} f(x, y) = \max_{y \in Y} \min_{x \in X} f(x, y)$$

- Min-max optimal point (x, y) is essentially unique (unique if f is strictly convex-concave, o.w. a convex set of solutions); value always unique
- E.g. $f(x, y) = x^2 - y^2 + x \cdot y$

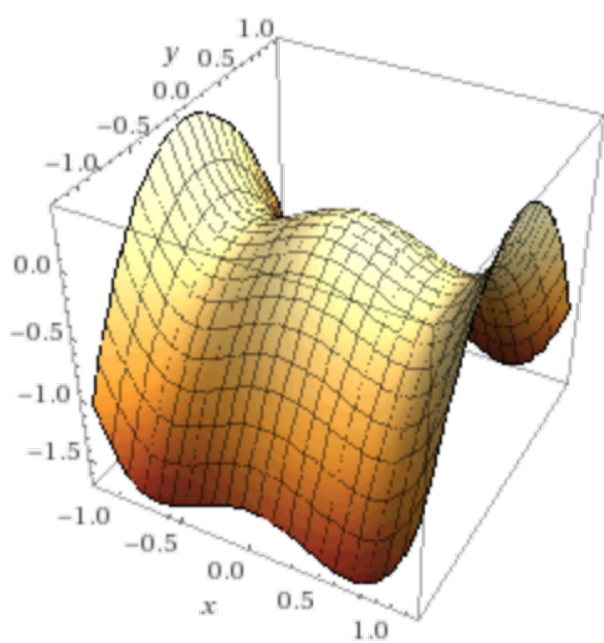


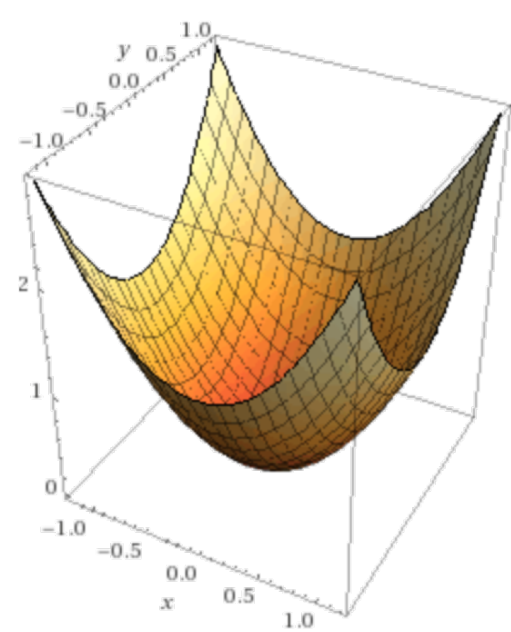
The Min-Max Theorem

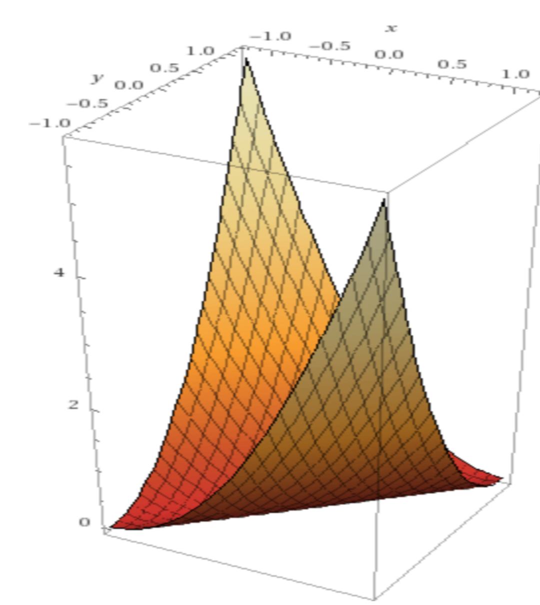
- **[von Neumann 1928]:** If $X \subset \mathbb{R}^n, Y \subset \mathbb{R}^m$ are compact and convex, and $f: X \times Y \to \mathbb{R}$ is convex-concave (i.e. $f(x, y)$ is convex in x for all y and is concave in y for all x), then

$$\min_{x \in X} \max_{y \in Y} f(x, y) = \max_{y \in Y} \min_{x \in X} f(x, y)$$

- Min-max optimal point (x, y) is essentially unique (unique if f is strictly convex-concave, o.w. a convex set of solutions); value always unique
- If f is not convex concave all bets are off
  - $f(x, y) = x^4 - x^2 - y^2$ $\qquad f(x, y) = x^2 + y^2$ $\qquad f(x, y) = (x - y)^2$







The Min-Max Theorem

- **[von Neumann 1928]:** If $X \subset \mathbb{R}^n, Y \subset \mathbb{R}^m$ are compact and convex, and $f: X \times Y \to \mathbb{R}$ is convex-concave (i.e. $f(x, y)$ is convex in x for all y and is concave in y for all x), then
$$\min_{x \in X} \max_{y \in Y} f(x, y) = \max_{y \in Y} \min_{x \in X} f(x, y)$$
- Min-max optimal point (x, y) is essentially unique (unique if f is strictly convex-concave, o.w. a convex set of solutions); value always unique
- Min-max points = equilibria of zero-sum game where min player pays max player $f(x, y)$
- von Neumann: "*As far as I can see, there could be no theory of games … without that theorem … I thought there was nothing worth publishing until the Minimax Theorem was proved*“
- When f is bilinear, i.e. $f(x, y) = x^{\mathrm{T}} A y + b^T x + c^T y$ and *X*, *Y* polytopes
  - **[von Neumann-Dantzig 1947, Adler IJGT’13]:** Minmax ⇔ strong LP duality
  - min-max solutions can be found w/ Linear Programming and vice versa
  - mathematical structure arguably crucial in recent success of computers beating humans in two-player zero-sum games (chess, poker, go)

The Min-Max Theorem (distributed dynamics *or very high dimensions*)

- **[Brown RAND'49]:** proposes *fictitious play* as a method to solve bilinear case on product of simplices:
$$\min_{x\in\Delta_n}\max_{y\in\Delta_m} x^T A y = \max_{y\in\Delta_m}\min_{x\in\Delta_n} x^T A y$$
- Fictitious play: $(x_t, y_t)_{t=1,\ldots}$ where for all t:
  - $x_t \in \mathrm{argmin} \sum_{\tau<t} f(\cdot, y_\tau)$
  - $y_t \in \mathrm{argmax} \sum_{\tau<t} f(x_\tau, \cdot)$
- **[Robinson Annals of Math'51]:** shows fictitious play converges in bilinear case in an average sense: $\frac{1}{t}\sum_\tau f(x_\tau, y_\tau) \to \min\max f(x,y)$
- **[Karlin'59]:** conjectures convergence rate is $\sim 1/\sqrt{t}$
- **[Daskalakis-Pan FOCS'14]:** actually exponentially slow $\left(\sim 1/t^{1/m+n}\right)$
- Faster methods?

Menu

- **Motivation**
- **Min-Max Theorem**
- No-Regret Learning and Online Convex Optimization
- Back to Min-Max Optimization

Menu

- **Motivation**
- **Min-Max Theorem**
- **No-Regret Learning and Online Convex Optimization**
- Back to Min-Max Optimization

Online Convex Optimization

- Game between learner and nature [min player's perspective in $\min_x \max_y f(x,y)$]
- Every day $t = 1, \ldots, T$:
  - Learner chooses $x_t \in X \subset \mathbb{R}^n$
  - World chooses convex function $f_t(\cdot)$ [in min-max problem $f_t(\cdot) \equiv f(\cdot, y_t)$]
  - Learner incurs loss $f_t(x_t)$; observes $f_t(\cdot)$
- **Learner's goal:**
  - $\cancel{\frac{1}{T}\sum_t f_t(x_t) \approx \frac{1}{T}\sum_t \min f_t(\cdot)}$ Unattainable (see notes)
  - $\frac{1}{T}\sum_t f_t(x_t) \approx \frac{1}{T}\min \sum_t f_t(\cdot)$ attainable [and sufficient for min-max]
  - $\frac{1}{T}\sum_t f_t(x_t) - \frac{1}{T}\min \sum_t f_t(\cdot)$: average regret of the learner
- **Theorem:** Suppose, $\forall t = 1, \ldots, T$, f_t is convex and L-Lipschitz. There exists learning algorithm such that $\frac{1}{\mathrm{T}}\sum_t f_t(x_t) - \min \frac{1}{\mathrm{T}}\sum_t f_t(\cdot) \le O_X\left(\frac{L}{\sqrt{T}}\right)$

  No-regret property (means avg regret$\rightarrow$ 0)

How to achieve no regret?

Setting: Every day $t = 1, \ldots, T$:
- learner chooses $x_t \in X$
- world chooses L-Lipschitz convex f'n $f_t(\cdot)$
- learner loses $f_t(x_t)$; observes $f_t(\cdot)$

Goal:
$\frac{1}{T}\sum_t f_t(x_t) - \frac{1}{T}\min \sum_t f_t(\cdot) \to 0$

- **Idea 1:** follow-the-leader (FTL): on day t choose $x_t \in \arg\min \sum_{\tau<t} f_\tau(\cdot)$
  - Average regret doesn't go to 0 ☹ [see notes]
  - **Issue:** overfitting
    - learner's actions move around abruptly
- **Idea 2:** regularize!
- follow-the-regularized-leader (FTRL): on day t choose

$$x_t \in \arg\min \left[\sum_{\tau<t} f_\tau(\cdot) + \frac{1}{\eta} \cdot R(\cdot)\right]$$

for some η and strongly convex regularization function $R(\cdot)$

Follow-the-Regularized Leader (FTRL)

Setting: Every day $t = 1, \ldots, T$:
- learner chooses $x_t \in X$
- world chooses L-Lipschitz convex f'n $f_t(\cdot)$
- learner loses $f_t(x_t)$; observes $f_t(\cdot)$

Goal:
$\frac{1}{T}\sum_t f_t(x_t) - \frac{1}{T}\min \sum_t f_t(\cdot) \to 0$

- **Def:** $R: X \to \mathbb{R}$ is α-strongly convex w.r.t. norm $\|\cdot\|$ iff for all $x, x_0 \in X$:

$$R(x) \geq R(x_0) + \nabla R(x_0)^T \cdot (x - x_0) + \frac{\alpha}{2}\|x - x_0\|^2$$

e.g.1: $R(x) = x^2/2$
e.g.2: $R(x) = -H(x)$, $x \in [0,1]$

- **FTRL:** On day t choose: $x_t \in \arg\min \left[\sum_{\tau<t} f_\tau(\cdot) + \frac{1}{\eta} \cdot R(\cdot)\right]$, for some parameter η, and some strongly convex regularization function $R(\cdot)$

- **Theorem:** Suppose, $\forall t = 1, \ldots, T$, f_t is convex and L-Lipschitz w.r.t. some norm $\|\cdot\|$, and R is 1-strongly convex w.r.t. $\|\cdot\|$. Then FTRL with parameter η satisfies:

$$\sum_t f_t(x_t) - \min \sum_t f_t(\cdot) \leq \frac{\max_X R(\cdot) - \min_X R(\cdot)}{\eta} + \eta \cdot L^2 \cdot T$$

- set $\eta = L^{-1} \cdot \sqrt{(\max R(\cdot) - \min R(\cdot))/T}$ to balance terms on RHS, and get average regret of $L \cdot \sqrt{(\max R(\cdot) - \min R(\cdot))/T}$

Follow-the-Regularized Leader (FTRL)

Setting: Every day $t = 1, \ldots, T$:
- learner chooses $x_t \in X$
- world chooses L-Lipschitz convex f'n $f_t(\cdot)$
- learner loses $f_t(x_t)$; observes $f_t(\cdot)$

Goal:
$\frac{1}{T}\sum_t f_t(x_t) - \frac{1}{T}\min \sum_t f_t(\cdot) \to 0$

- **Def:** $R: X \to \mathbb{R}$ is α-strongly convex w.r.t. norm $\|\cdot\|$ iff for all $x, x_0 \in X$:

$$R(x) \geq R(x_0) + \nabla R(x_0)^T \cdot (x - x_0) + \frac{\alpha}{2}\|x - x_0\|^2$$

e.g.1: $R(x) = x^2/2$
e.g.2: $R(x) = -H(x)$, $x \in [0,1]$

- **FTRL:** On day t choose: $x_t \in \arg\min\left[\sum_{\tau<t} f_\tau(\cdot) + \frac{1}{\eta} \cdot R(\cdot)\right]$, for some parameter η, and some strongly convex regularization function $R(\cdot)$
- FTRL special cases:
  - FTRL w/ ℓ_2^2-regularizer $\approx$ online gradient descent [notes]
  - FTRL on simplex w/ negative entropy regularizer $=$ multiplicative-weights-update method

FTRL and Min-Max

- Suppose $f(x, y)$ convex-concave, and both x and y players run FTRL
- Namely:
  - the x-player chooses x_t by applying FTRL to observed losses $f(\cdot, y_t)$
  - the y-player chooses y_t by applying FTRL to observed losses $-f(x_t,\cdot)$
- **Theorem:** If x and y player play as above, then:
  - $\frac{1}{T}\sum_{t=1}^{T} f(x_t, y_t) = \min_x \max_y f(x, y) \pm O(\frac{1}{\sqrt{T}})$
  - Moreover, the average strategies $\bar{x}_T = \frac{1}{T}\sum_t x_t$ and $\bar{y}_T = \frac{1}{T}\sum_t y_t$ are a $O(\frac{1}{\sqrt{T}})$-approximate Nash equilibrium, i.e.
    - $f(\bar{x}_T, \bar{y}_T) \leq \min f(\cdot, \bar{y}_T) + O(\frac{1}{\sqrt{T}})$
    - $f(\bar{x}_T, \bar{y}_T) \geq \max f(\bar{x}_T, \cdot) - O(\frac{1}{\sqrt{T}})$
- Proof: notes

Menu

- **Motivation**
- **Min-Max Theorem**
- **No-Regret Learning and Online Convex Optimization**
- Back to Min-Max Optimization

Menu

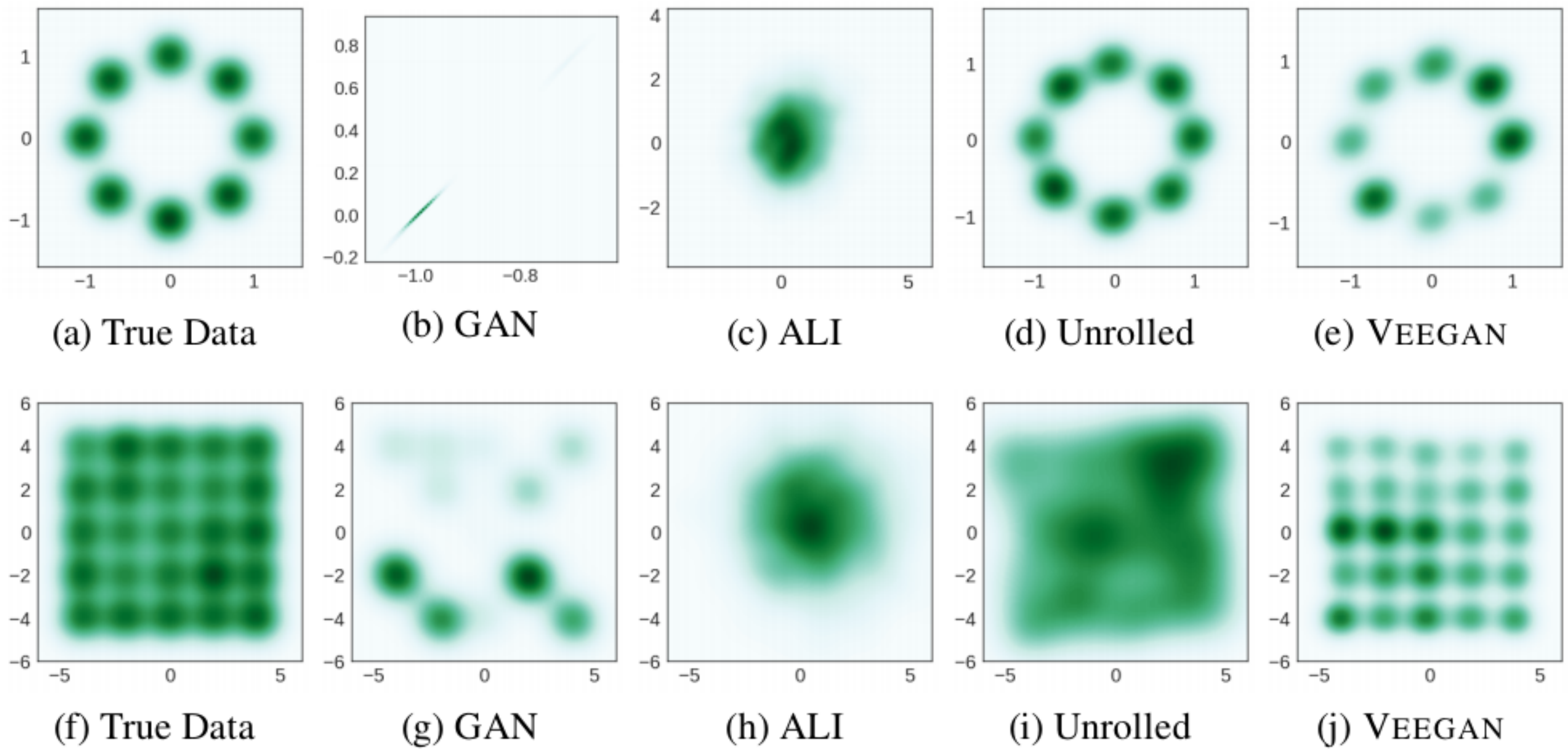
- **Motivation**
- **Min-Max Theorem**
- **No-Regret Learning and Online Convex Optimization**
- **Back to Min-Max Optimization**

Challenges in GAN Training

- Recall, they are defined by setting up a ***game*** between a *Generator* deep NN, w/ parameters θ_g, and a *Discriminator* deep NN, w/ parameters θ_d:
$$\inf_{\theta_g} \sup_{\theta_d} \big(f(\theta_g, \theta_d) \big)$$
- **Training**: generator and discriminator run online gradient descent and ascent respectively and in parallel to update their parameters θ_g, θ_d
- **Question:** will paired online gradient descent/ascent style dynamics converge? to what?
- **Challenge 1:** objective function $f(\theta_g, \theta_d)$ isn't convex-concave
  - So what is the goal?
    1. **[Daskalakis-Panageas NeurIPS'18]** study local saddles (don't necessarily exist)
    2. **[Jin-Netrapali-Jordan'19]** study local min of $\max_y f(\cdot, y)$ function (exist under mild conditions)
       - E.g. $\min_{x \in [0,1]} \max_{y \in [0,1]} -(x-y)^2$: 1 doesn't exist, 2 does exist
  - Are above reasonable? Well,…
    - under 1: maybe my trained discriminator cannot locally improve discrimination, but some other discriminator (e.g. your brain) can discriminate really well between real and generated images (locally optimal discrimination isn't sufficient)
    - under 1 and 2: if my trained discriminator is optimal and my trained generator is locally optimal, it might just have given up

Mode Collapse

Figure 2: Density plots of the true data and generator distributions from different GAN methods trained on mixtures of Gaussians arranged in a ring (top) or a grid (bottom).

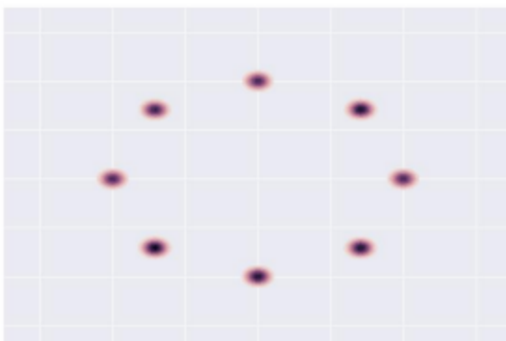


VEEGAN: Reducing Mode Collapse in GANs using Implicit Variational Learning
[Akash Srivastava](), [Lazar Valkov](), [Chris Russell](), [Michael U. Gutmann](), [Charles Sutton]()

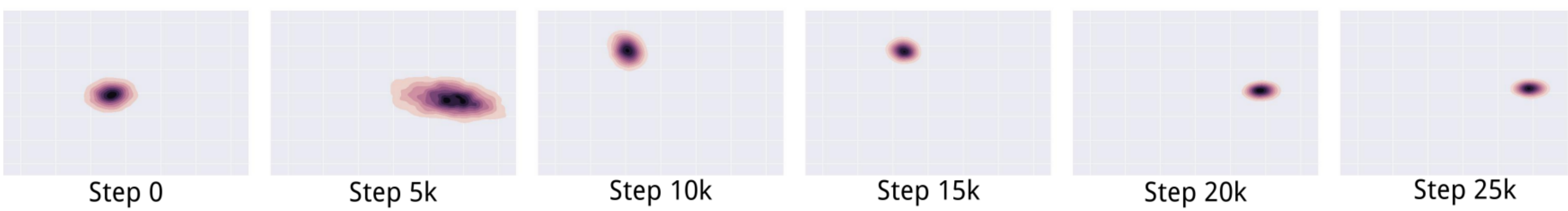
Challenges in GAN Training

- Recall, they are trained by setting up a ***game*** between a *Generator* deep NN, w/ parameters θ_g, and a *Discriminator* deep NN, w/ parameters θ_d:
$$\inf_{\theta_g} \sup_{\theta_d} \big(f(\theta_g, \theta_d)\big)$$
- **Training**: generator and discriminator run gradient descent and ascent respectively to update their parameters θ_g, θ_d
- **Question:** even ignoring expectation approximation errors, will paired gradient descent/ascent style dynamics converge? to what?
- **Challenge 2:** oscillations
  - *even if $f(\theta_g, \theta_d)$ is convex-concave*, we only argued that gradient/descent ascent, or FTRL *converge in an **average sense***
  - i.e. $\overline{\theta_g}_T = \frac{1}{T}\sum_t \theta_{g_t}$ and $\overline{\theta_d}_T = \frac{1}{T}\sum_t \theta_{d_t}$ would be an approximate saddle but we didn't provide any guarantees for the last iterate $(\theta_{g_T}, \theta_{d_T})$…
  - and there aren't such guarantees generically

Training Oscillations: Gaussian Mixture



True Distribution: Mixture of
8 Gaussians on a circle

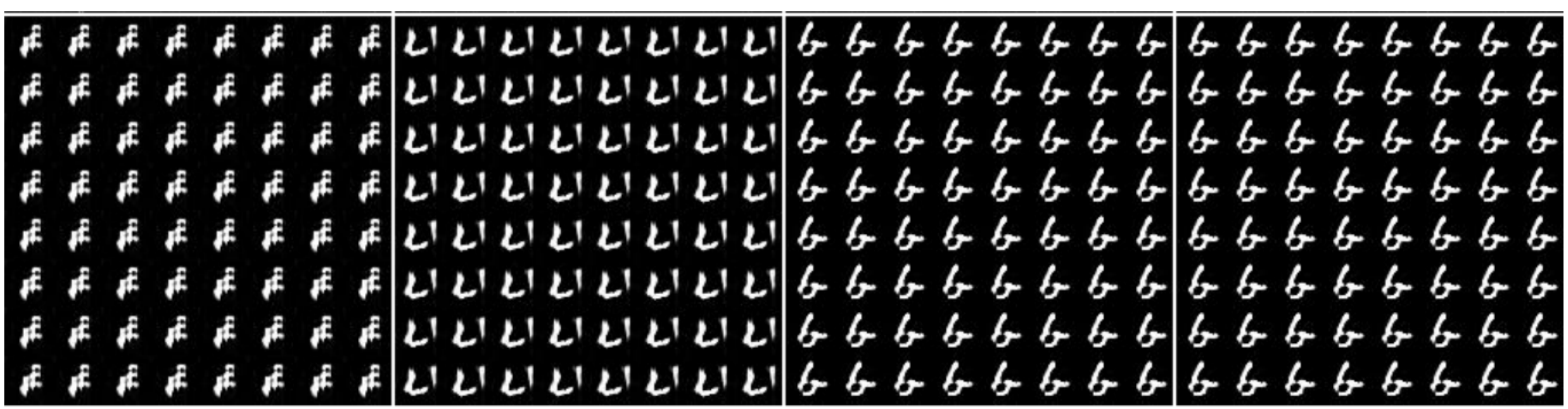


Output Distribution of standard GAN, trained via gradient descent/ascent dynamics:
cycling through modes at different steps of training

from **[Metz et al ICLR’17]**

Training Oscillations: Handwritten Digits

True Distribution: MNIST



10k steps 20k steps 50K steps 100k steps

Output Distribution of standard GAN, trained via gradient descent/ascent dynamics *cycling through "proto-digits" at different steps of training*

from **[Metz et al ICLR'17]**

Training Oscillations: even for bilinear objectives!

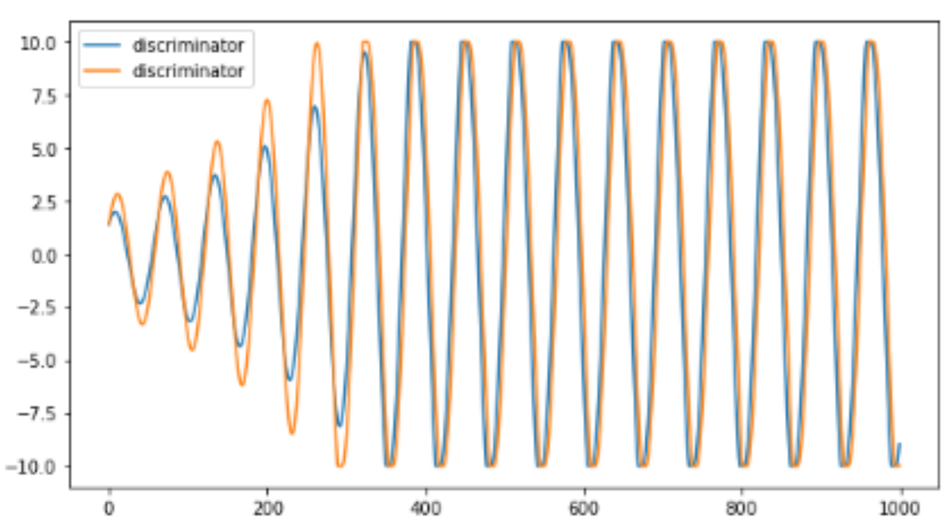
- **True distribution:** isotropic Normal distribution, namely $X \sim \mathcal{N}\left(\begin{bmatrix}3\\4\end{bmatrix}, I_{2\times 2}\right)$
- **Generator architecture**: $G_{\boldsymbol{\theta}}(Z) = \boldsymbol{\theta} + Z$ (adds input Z to internal params)
- **Discriminator architecture**: $D_{\boldsymbol{w}}(\cdot) = \langle \boldsymbol{w}, \cdot \rangle$ (linear projection)

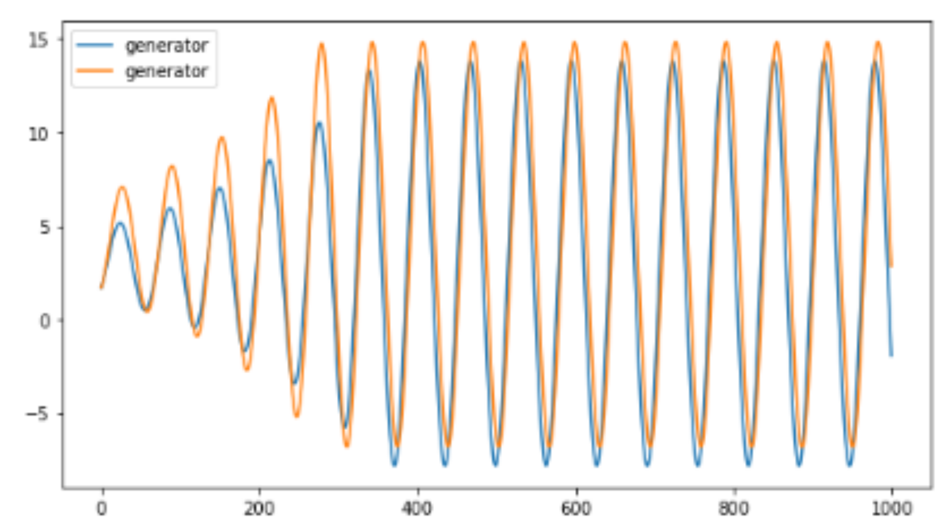
Z, θ, w: 2-dimensional

- **W-GAN objective:** $\min_{\boldsymbol{\theta}} \max_{\boldsymbol{w}} \mathbb{E}_X[D_{\boldsymbol{w}}(X)] - \mathbb{E}_Z[D_{\boldsymbol{w}}(G_{\boldsymbol{\theta}}(Z))]$

$$= \min_{\boldsymbol{\theta}} \max_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}} \cdot \left(\begin{bmatrix}3\\4\end{bmatrix} - \boldsymbol{\theta}\right)$$

convex-concave function

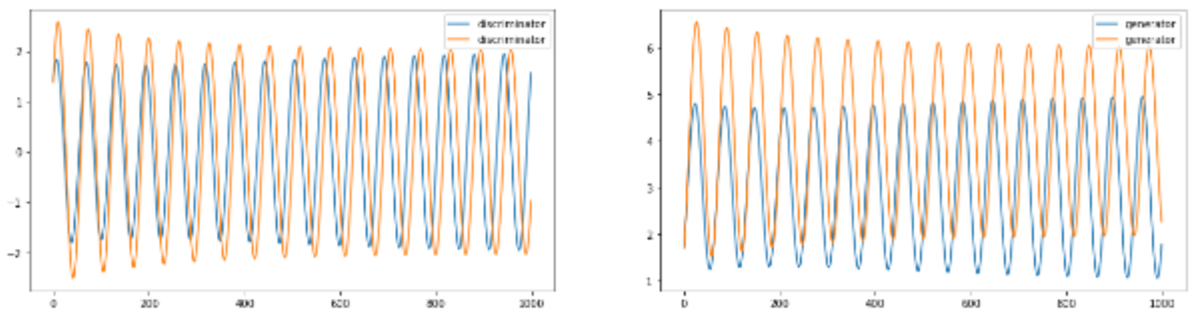




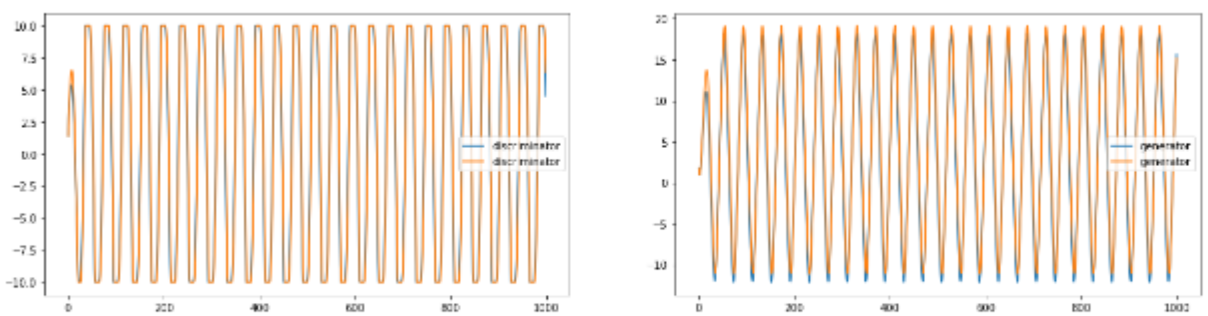
Gradient Descent Dynamics

from **[Daskalakis, Ilyas, Syrgkanis, Zeng ICLR'18]**

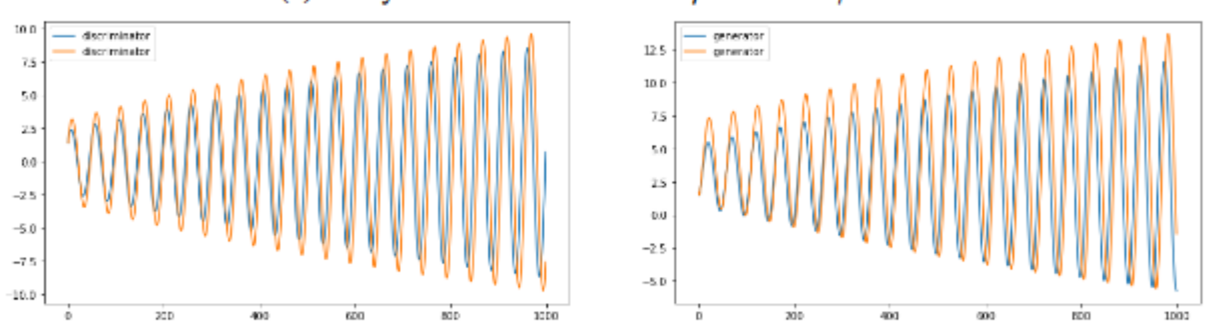
Training Oscillations:
persistence under many variants of Gradient Descent



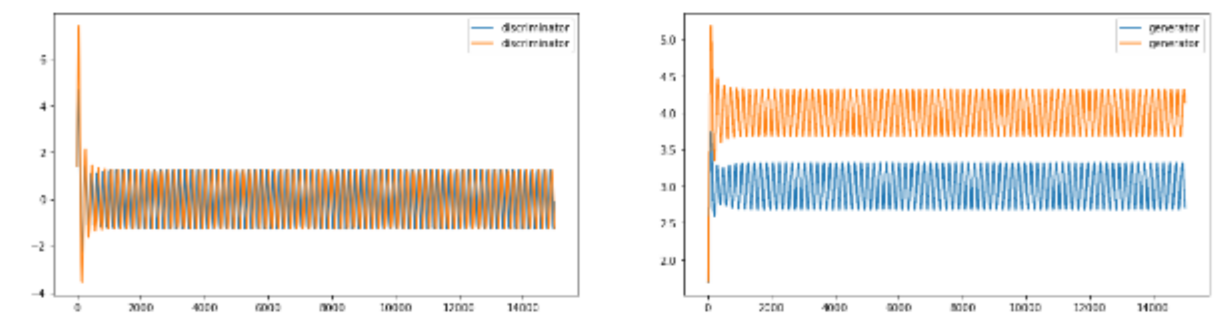
(a) GD dynamics with a gradient penalty added to the loss. $\eta = 0.1$ and $\lambda = 0.1$.



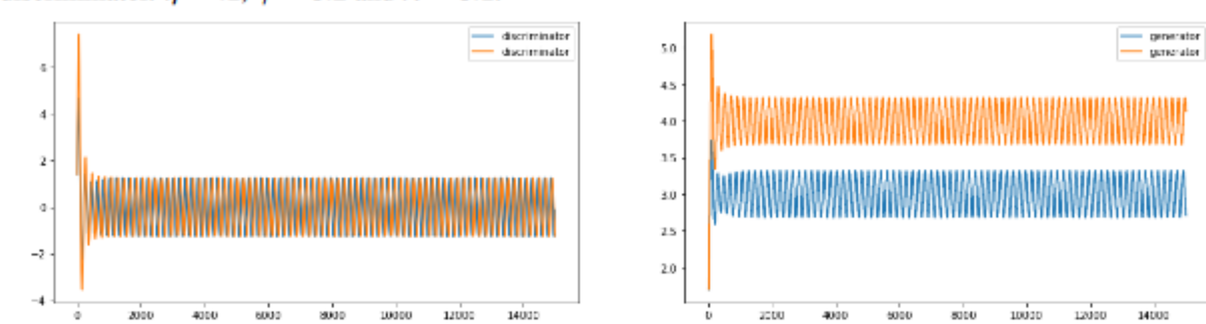
(b) GD dynamics with momentum. $\eta = 0.1$ and $\gamma = 0.5$.



(c) GD dynamics with momentum and gradient penalty. $\eta = .1$, $\gamma = 0.2$ and $\lambda = 0.1$.



(d) GD dynamics with momentum and gradient penalty, training generator every 15 training iterations of the discriminator. $\eta = .1$, $\gamma = 0.2$ and $\lambda = 0.1$.



(e) GD dynamics with Nesterov momentum and gradient penalty, training generator every 15 training iterations of the discriminator. $\eta = .1$, $\gamma = 0.2$ and $\lambda = 0.1$.

from **[Daskalakis, Ilyas, Syrgkanis, Zeng ICLR'18]**

Menu

- **Motivation**
- **Min-Max Theorem**
- **No-Regret Learning and Online Convex Optimization**
- **Back to Min-Max Optimization**
  - **Last Iterate Convergence**

Gradient Descent w/ Negative Momentum

- Variant of gradient descent:

$$\forall t:\ x_{t+1} = x_t - \eta \cdot \nabla f(x_t) + \boldsymbol{\eta/2 \cdot \nabla f(x_{t-1})}$$

- **Interpretation:** undo today, some of yesterday's gradient; ie negative momentum

- Gradient Descent w/ negative momentum

  = **Optimistic** FTRL w/ ℓ_2^2-regularization **[Rakhlin-Sridharan COLT'13, Syrgkanis et al. NeurIPS'15]**

  = unconstrained **Popov's method** **[Popov 1980]**

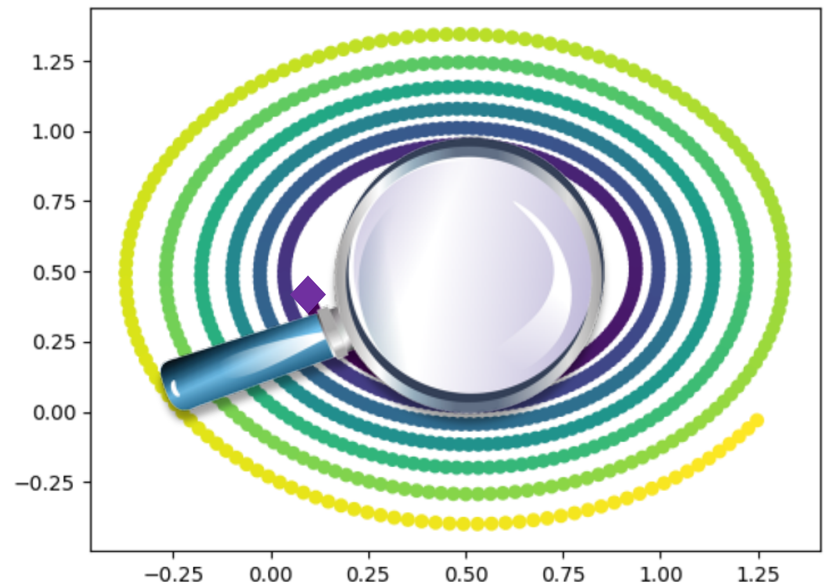
  $\approx$ **extra-gradient** method **[Korpelevich'76, Chiang et al COLT'12, Mertikopoulos et al'18]**

  $=$ **mirror prox** method w/ ℓ_2^2-regularization **[Nemirovski'04, Mohtari-Ozdaglar-Pattathil'19]**

- **Does it help in min-max optimization?**

Negative Momentum: why it could help

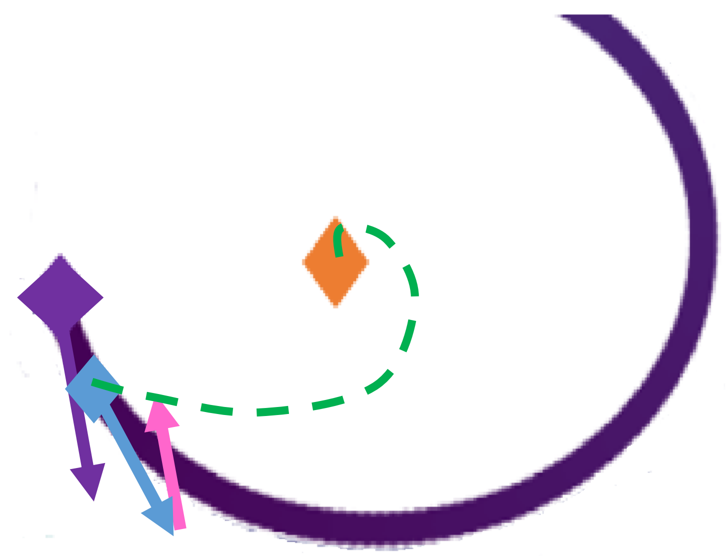
- E.g. $f(x, y) = (x - 0.5) \cdot (y - 0.5)$



$$\boxed{\begin{aligned} x_{t+1} &= x_t - \eta \cdot \nabla_x f(x_t, y_t) \\ y_{t+1} &= y_t + \eta \cdot \nabla_y f(x_t, y_t) \end{aligned}}$$

◆ : start

◆ : min-max equilibrium



$$\boxed{\begin{aligned} x_{t+1} &= x_t - \eta \cdot \nabla_x f(x_t, y_t) \\ &\qquad + \boldsymbol{\eta/2 \cdot \nabla_x f(x_{t-1}, y_{t-1})} \\ y_{t+1} &= y_t + \eta \cdot \nabla_y f(x_t, y_t) \\ &\qquad - \boldsymbol{\eta/2 \cdot \nabla_y f(x_{t-1}, y_{t-1})} \end{aligned}}$$

Negative Momentum: convergence

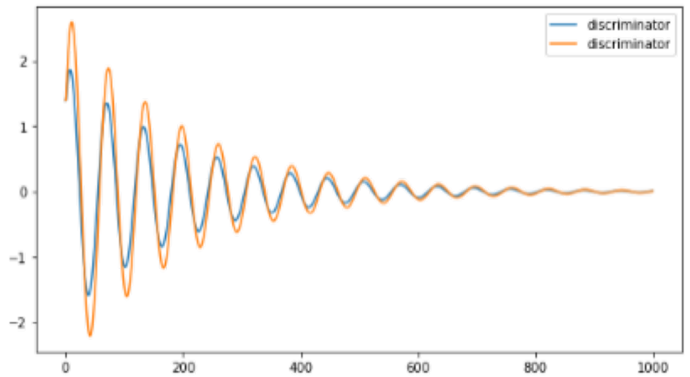
- **Optimistic gradient descent-ascent (OGDA)** dynamics:

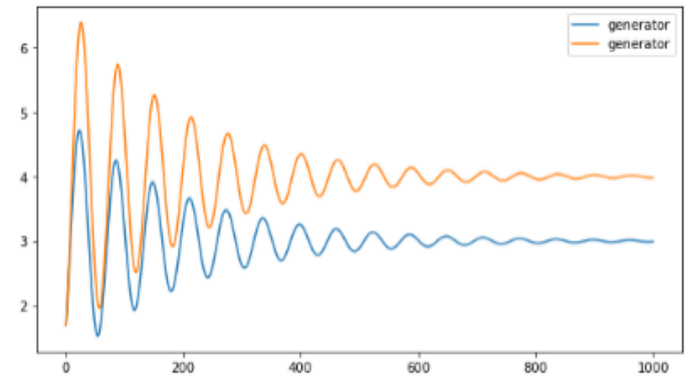
$$\forall t:\ x_{t+1} = x_t - \eta \cdot \nabla_x f(x_t, y_t) + \frac{\eta}{2} \cdot \nabla_x f(x_{t-1}, y_{t-1})$$
$$y_{t+1} = y_t + \eta \cdot \nabla_y f(x_t, y_t) - \frac{\eta}{2} \cdot \nabla_y f(x_{t-1}, y_{t-1})$$

- **[Daskalakis-Ilyas-Syrgkanis-Zeng ICLR'18]:** **OGDA** exhibits last iterate convergence & fast rates for *unconstrained* bilinear games: $\min_{x \in \mathbb{R}^n} \max_{y \in \mathbb{R}^m} f(x,y) = x^{\mathrm{T}} A y + b^T x + c^T y$

- **[Liang-Stokes AISTATS'19, Gidel et al AISTATS'19]:** …convergence rate is geometric if A is well-conditioned, extends to strongly convex-concave functions $f(x,y)$

- E.g. in previous isotropic Gaussian case: $X \sim \mathcal{N}\big((3,4), I_{2\times 2}\big)$, $G_\theta(Z) = \theta + Z$, $D_w(\cdot) = \langle w, \cdot \rangle$





Negative Momentum: convergence

- **Optimistic gradient descent-ascent (OGDA)** dynamics:

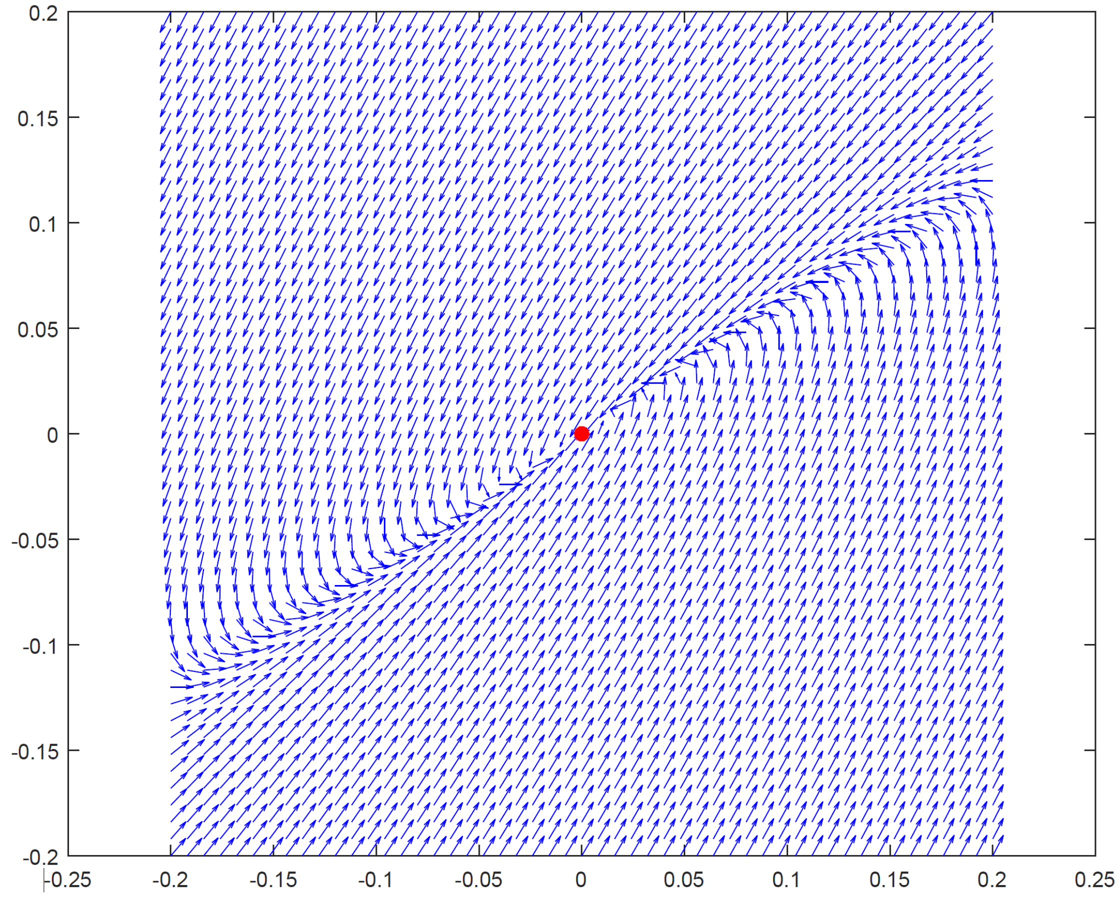
$$\forall t:\ x_{t+1} = x_t - \eta \cdot \nabla_x f(x_t, y_t) + \frac{\eta}{2} \cdot \nabla_x f(x_{t-1}, y_{t-1})$$
$$y_{t+1} = y_t + \eta \cdot \nabla_y f(x_t, y_t) - \frac{\eta}{2} \cdot \nabla_y f(x_{t-1}, y_{t-1})$$

- **[Daskalakis-Ilyas-Syrgkanis-Zeng ICLR'18]:** OGDA exhibits last iterate convergence & fast rates for *unconstrained* bilinear games: $\min_{x \in \mathbb{R}^n} \max_{y \in \mathbb{R}^m} f(x, y) = x^{\mathrm{T}} A y + b^T x + c^T y$

- **[Liang-Stokes AISTATS'19, Gidel et al AISTATS'19]:** …convergence rate is geometric if A is well-conditioned, extends to strongly convex-concave functions $f(x, y)$
- **[Mohtari et al'19]:** …ditto for extra-gradient, mirror-prox methods
- **[Daskalakis-Panageas ITCS'19]:** ***Projected*** OGDA exhibits last iterate convergence even for *constrained* bilinear games: $\min_{x \in \Delta_n} \max_{y \in \Delta_m} x^{\mathrm{T}} A y$ = all linear programming
- **General Comment:** asymptotic convergence results were known already by Korpelevich and Popov for extragradient and negative momentum respectively
- **[w/ Jelena Diakonikolas, Mike Jordan]:** results for general constraints + convergence rates + general Bregman divergences

Negative Momentum: in the Wild

- Can try optimism for non convex-concave min-max objectives $f(x, y)$
- **Issue [Daskalakis, Panageas NeurIPS'18]:** No hope that stable points of **OGDA** or GDA are only local min-max points
- e.g. $f(x, y) = -\frac{1}{8} \cdot x^2 - \frac{1}{2} \cdot y^2 + \frac{6}{10} \cdot x \cdot y$



Gradient Descent-Ascent field

- Nested-ness: Local Min-Max $\subseteq$ Stable Points of GDA $\subseteq$ Stable Points of **OGDA**
- (stability refers to linear stability and left inclusion for strong local min-max points)

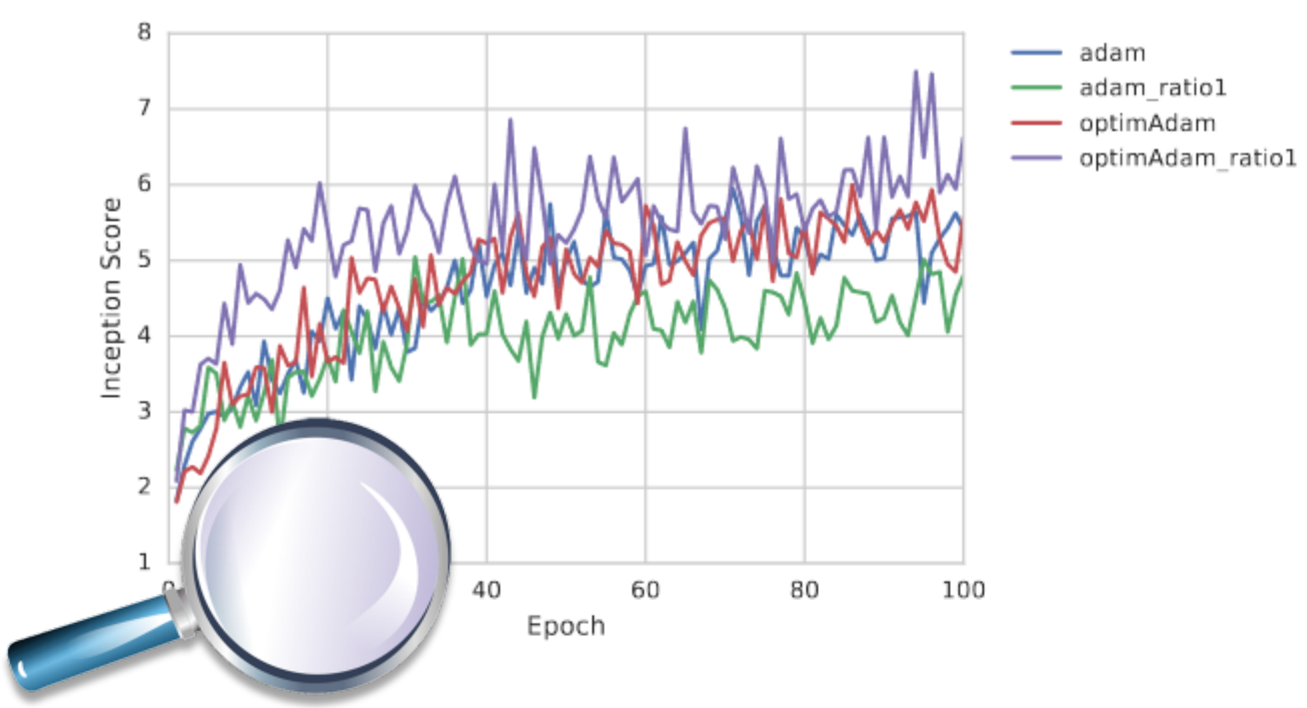
Negative Momentum: in the Wild

- Can try optimism for non convex-concave min-max objectives $f(x, y)$
- **Issue [Daskalakis, Panageas NeurIPS'18]:** No hope that stable points of **OGDA** or GDA are only local min-max points
    - Local Min-Max $\subseteq$ Stable Points of GDA $\subseteq$ Stable Points of **OGDA**
- also **[Adolphs et al. 18]:** left inclusion
- **Question:** identify first-order method converging to local min-max w/ probability 1

- While this is pending, evaluate optimism in practice…
- **[Daskalakis-Ilyas-Syrgkanis-Zeng ICLR'18]:** propose ***optimistic Adam***
    - ***Adam***, a variant of gradient descent proposed by **[Kingma-Ba ICLR'15]**, has found wide adoption in deep learning, although it doesn't always converge **[Reddi-Kale-Kumar ICLR'18]**
    - ***Optimistic Adam*** is the right adaptation of Adam to "undo some of the past gradients"

Optimistic Adam on CIFAR10

- Compare Adam, **Optimistic Adam**, trained on CIFAR10, in terms of Inception Score
- No fine-tuning for **Optimistic Adam**, used same hyper-parameters for both algorithms as suggested in Gulrajani et al. (2017)



Optimistic Adam on CIFAR10

- Compare Adam, **Optimistic Adam**, trained on CIFAR10, in terms of Inception Score
- No fine-tuning for **Optimistic Adam**, used same hyper-parameters for both algorithms as suggested in Gulrajani et al. (2017)

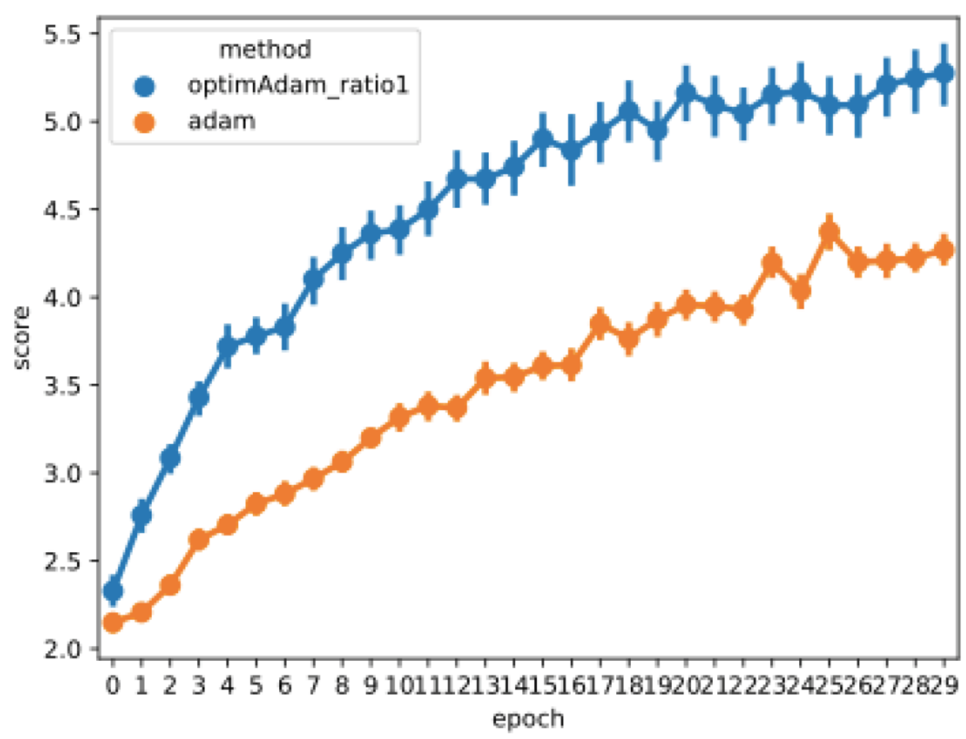
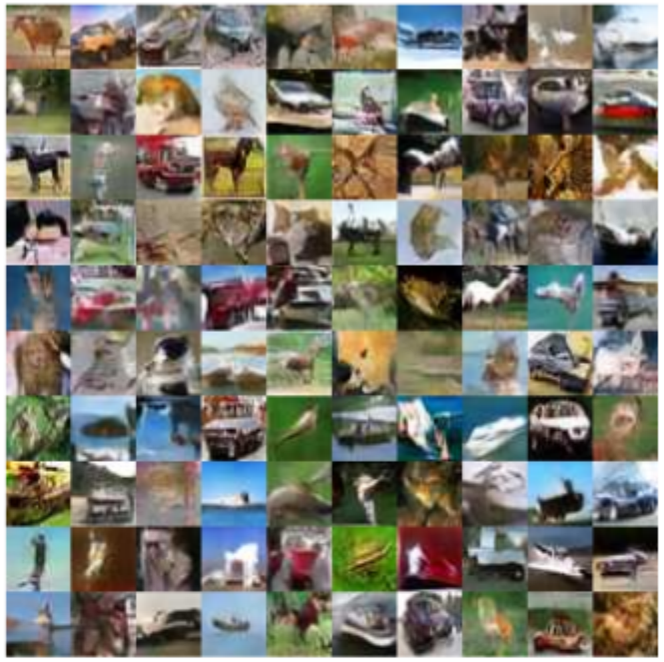


Figure 14: The inception scores across epochs for GANs trained with Optimistic Adam (ratio 1) and Adam (ratio 5) on CIFAR10 (the two top-performing optimizers found in Section 6, with 10%-90% confidence intervals. The GANs were trained for 30 epochs and results gathered across 35 runs.



(b) Sample of images from Generator of Epoch 94, which had the highest inception score.

- Further supporting evidence for negative momentum methods by **[Gidel et al. AISTATS'19]**